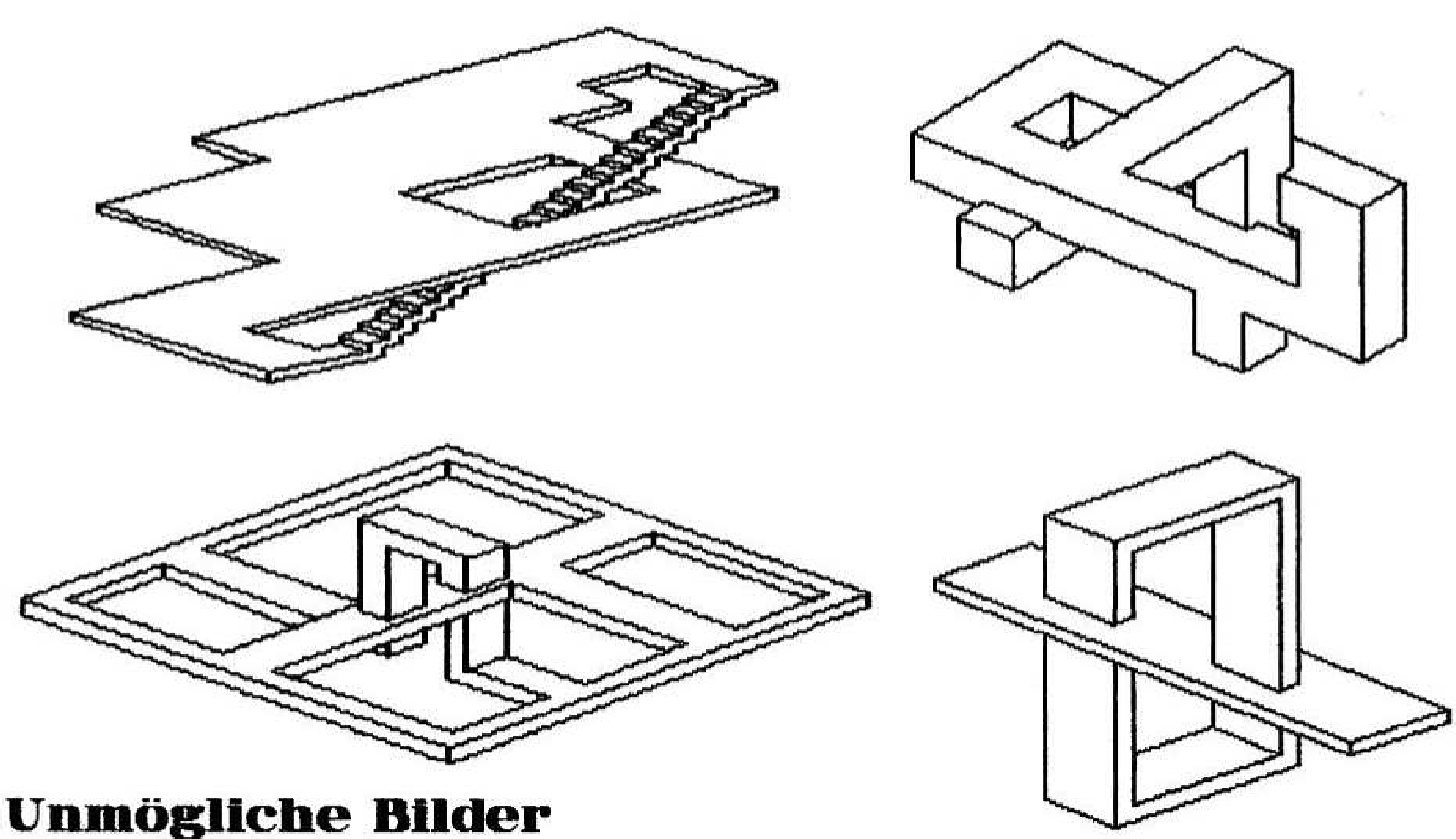
Spectrum Profi Club

für alle Spectrum und SAM Freunde



Unmögliche Bilder

Wolfgang und Monika Haller
Ernastr. 33, 51069 Köln, Tel. 0221/685946
Bankverbindung: Dellbrücker Volksbank
BLZ 370 604 26, Konto-Nr. 7404 172 012

INFO

Oktober 1994

Smalltalk

Schon wieder ist ein Monat vorbei, und an den Temperaturen kann man sehen, wie schnell wir doch jetzt dem Winter entgegensteuern. Dabei ist es noch garnicht solange her, als wir über die große Hitze gestöhnt haben.
Immerhin scheint das Wetter euch wieder mehr vor den Compi zu treiben, was die Mitarbeit am Info zeigt. Etwas, was uns immer sehr freut.
Erstaunliches geschieht auch sonst: Wir erhalten immer mehr Post aus dem Ausland, vornehmlich England. Diesmal haben wir gleich zwei Artikel von Miles Kinloch übersetzt und ins Info gebracht, und das ist nur ein Teil von dem, was er uns geschickt hat. Miles ist durch "Alchemist" auf unseren SPC aufmerksam geworden. Thank you, Miles, your infos are very useful and welcome.
Nun sind wir auch im Kontakt mit dem dänischen Sinclair-Club "Sinclair Freak'eren", dessen Redaktionsmitglied Preben Sörensen seit vorigem Monat ja auch Mitglied bei uns ist. Sie bringen ein vierteljährliches Info heraus, ich muß jedoch gestehen, das die sprachliche Barriere sehr groß ist. Der Club vertritt nicht nur den Spectrum, sondern auch noch stark den QL, aber auch auf den SAM bin ich in diesem Heft gestossen. Mal schauen, wie es weitergeht.
Ein ganz positives Echo erhielten wir von Amanda, dem Star des letzten Info-Titels. Das "Knuckle-Girl" hat bereits zwei weitere Demos angekündigt, auf die wir schon gespannt warten. "Amy" ist Techno Musikfan, mal sehen, wie sich dies auf ihre Soundtracker Musik niederschlägt. Mehr dazu, wenn wir die neuen Demos haben. Dasgleiche gilt auch für "Sinclair Classic 8".
Heute wollen wir auch schon einmal einen Ausblick auf das kommende Info wagen. Vorgesehen ist ein Artikel für eine externe Tastatur, etwas, wovon viele Spectrum-User bekanntlich träumen. Außerdem solltet ihr wieder die ominöse Postkarte vorfinden, die euch wie jedes Jahr an die Verlängerung eurer Mitgliedschaft für 1995 erinnern soll.
Noch eine Bitte: Zur Zeit brauchen wir keine weiteren MOD-Files mehr. Vielen Dank an alle, die uns bisher welche gesandt haben, aber wir haben noch nicht einmal alle hören und sichten können. Wir melden uns, sobald wieder Bedarf besteht.
Und nun wieder viel Spaß am Info. Euer WoMo-Team.

Es ist unsere traurige Pflicht, das Ableben eines unserer Mitglieder bekanntzugeben.

In Memoriam

Heinz Kober

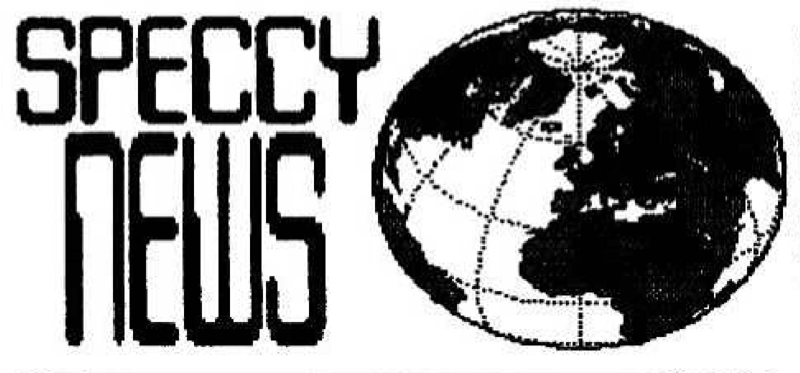
Mitglieds-Vorstellung

Hallo Spectrum-User!
Seit November 93 bin ich Mitglied im SPC. Da meine große Leidenschaft 8-Bit Heimcomputer sind und ich auch über ein paar Speccy-Brocken verfüge, ist die Mitgliedschaft für mich Pflicht. Ich bin auch Mitglied im ABBUC, im SHARP-User Club und im ZX-TEAM, dort besonders aktiv. Durch mehrfache Mitgliedschaften ensteht zwischen den Clubs eine, wie ich meine, wünschenswerte Verflechtung.
Ich bin Baujahr 1948, fast 25 Jahre verheiratet und habe 2 Kinder (20/24). Zur Zeit bin ich noch dabei, meinen Spectrum-Bestand aufzubauen, ich übe gerade IF1-Reparatur (Saumühsam...). Von Beruf bin ich E-Techniker/E-Meister und repariere weltweit Schiffselektronik, bin also telefonisch nicht immer zu erreichen. Ich fühle mich im SPC sehr gut aufgehoben und betreut und möchte an dieser Stelle ganz besonders WoMo und Jean Austermühle für die mehrfache Hilfe danken, ist einfach riesig!
An dieser Stelle habe ich auch gleich eine Bitte an euch alle: Es gibt einen Spectrum-Emulator auf dem Sharp MZ 800 (einem auch schon recht betagten 8-Bitter auf Z80-Basis, nach eigenen Angaben etwas für Museumsinformatiker), der im MZ-Info Heft 7 genau beschrieben wurde. Nach mühevoller Kleinarbeit hat man sogar die Tastenbelegungen herausgefunden und ein Demo-programm gestartet. Nun sind natürlich alle Sharpies dazu aufgerufen worden, in ihrer Nähe nach Sinclair-Usern Ausschau zu halten und Sinclair-Kassetten auszuprobieren! Der Autor dieses Artikels heißt Klaus Friese, und wer ihm von den Sharpies einmal schreiben möchte, hier seine Adresse:

Klaus Friese, Glindersweg 66
21029 Hamburg.

Das war es erst einmal, liebe Grüße an alle Speccy-User von Willi

Wilhelm Mannertz, Lindenstraße 12
24223 Raisdorf, Tel. 04307/6271



Multimedia-Show in Neu-Ulm

Am 05. November 94 in Neu-Ulm in der Mehrzweckhalle Gerlenhofen findet bereits zum dritten Male die Neu-Ulmer Multi-Media Messe statt. Diesmal erstmals mit einem Vertreter aus der Sinclair Riege. Der Spectrum-User-Club wird dort mit einem Stand vertreten sein. Erfreulich wäre es, wenn vielleicht das eine oder andere SPC-Mitglied vorbeischaut.
Also, Multi-Media-Messe, 05.11.94 in Neu-Ulm von 09.30 - 18.00 Uhr. Thomas Eberle

FORMAT präsentiert das neue DX1 Disk-IF

Nun wird es offiziell angeboten, das neue DX1 Disk Interface. Es ist für den Spectrum 48K, 128K, +2 und +2A erhältlich. Basierend auf dem bekannten und getesteten Plus D wird es mit einem eingebauten UNI-DOS System geliefert. Es erlaubt die Benutzung von einem oder zwei Laufwerken (3,5" oder 5,25") und ist kompatibel zu den meisten Druckern mit Parallel-Schnittstelle.
Der offizielle Preis beträgt 59,95 Pfund, für INDUG Mitglieder unter Angabe der Mitgliedsnummer jedoch nur 49,95 Pfund. Wer es für den +2A haben möchte, sollte dies unbedingt schriftlich vermerken.
Wer gleich ein 3,5" Laufwerk mitbestellen möchte, kann dies auch tun. Diese werden von Blue Alpha Electronics produziert und kosten ebenfalls 59,95 Pfund, wenn sie mit dem DX1 zusammen bestellt werden.
Für Briten kommen noch 2 Pfund für das DX1 und/oder 3,50 Pfund für das Laufwerk an Versandversicherung hinzu. Die Kosten für uns vom Festland werden auf Anfrage mitgeteilt. Bitte auch hier gegebenenfalls die INDUG Mitgliedschaftsnummer angeben.
Wer also interessiert ist, der schreibe (oder schicke direkt einen an FORMAT ausgestellten Scheck) an:

Format Publications (DX1)
34 Bourton Road, Gloucester
GL4 0LE, England

Sonderheft des SUC

Hallo Leute, ich muß sagen ich bin doch etwas entäuscht von den SPC Mitgliedern. Da habe ich euch ein Spitzen-Softwareangebot gemacht und kaum einer greift zu. Was ist los, eine Mark pro Spiel ist doch geschenkt!
Da hat es vielleicht auch wenig Zweck anzukündigen, das wir beim SUC jetzt Listen aller Soft- und Hardwarehändler gesammelt haben um daraus ein Sonderheft zu machen, in dem Bezugsadressen für fast alles, was es für den Spectrum gibt genannt werden. Vor allem im Bereich Software ist das Angebot riesig. Das Sonderheft kostet für Nicht-SUC-Mitglieder 2,00 DM Porto und Kopierkosten.
Also, falls doch jemand Interesse hat, schreibt an Spectrum-User-Club, Gastäckerstr. 23, 70794 Filderstadt.

From Russia with love

Von einer 5-wöchigen Urlaubsreise durch Ost-Europa brachte sich Roelof Koning von der SGG zwei Spectrum 48K Clones mit. Seine Beschreibung der beiden Geräte habe ich auf Platzgründen auf das wichtigste reduziert.
'THE BYTE': Rotes Gehäuse (!) mit richtiger Tastatur. Der Hersteller des 1993 hergestellten Gerätes ist unbekannt, ein Handbuch fehlt. Das Gerät trägt die Aufschrift: 'Pristavka Televisionaja Programmiroemaja' und hat die Seriennummer 895. Das Netzteil für 5V ist etwa 3mal so groß wie das Original.
An der Rückseite befinden sich folgende Anschlüsse: Tape DIN-Buchse, Video (RGB), ein Composite Videosignal (nur für Graustufen) und ein Kempston-Joystick Anschluß. Es fehlt ein 'richtiger' Bus, jedoch sind die meisten Verbindungen hinter einem Schlitz zu finden.
Das Innere sieht sehr nach reiner Handarbeit aus, alle Chips tragen russische Bezeichnungen. Besonderheit: 1 16K Eprom für das Sinclair ROM in originaler Form. Ersatz-ULA in der Größe einer Briefmarke mit 64 Verbindungen! Auch nach mehrstündigem Gebrauch bleiben die Chips überraschend kalt.
'THE SINTEZ': Sieht aus wie ein Spectrum-Plus. Die Tasten 'wackeln' etwas und sehen aus, als kämen sie aus verschiedenen Produktionen. Hersteller unbekannt, kein Handbuch und auch kein Netzteil, das Gerät läuft jedoch mit regulierten 5V genauso wie mit 4,8V.
An der Rückseite befinden sich: RGB Video Din-Buchse, Composite (nur für Grausstufen) und ein Kempston Joystick Anschluß. Die R, G und B Ausgänge können justiert werden. Auch hier ist kein Bus, sondern wieder ein Schlitz.
Im Inneren findet man 50 Standardchips, wahrscheinlich aus der bekannten 74er Familie, jedoch auch hier wieder mit russischen Bezeichnungen. Das CLOCK-Signal kommt von einem 9 MHz (!) Kristall und das ROM ist das gleiche wie das Sinclair Original.
Beide Geräte benutzen einen separaten Video-Speicher und die Z80 'WAIT' line ist intern mit den anderen Schaltkreisen verbunden. Es scheint, als ob die Speicherinhalte nicht Sinclairmäßig sind, und das die internen Timings vom normalen 48K Spectrum abweichen können.

DIE SEITEN FÜR DEN SAM!

Die IDE-Schnittstelle braucht ein DOS

Ich habe mit ein paar Leuten gesprochen, und diese finden die IDE Festplatten-Schnittstelle aus dem Juli-Infoheft eine tolle Entwicklung. Es ist mit nur 10 Chips ohne Zweifel ein sehr schönes Design, wirklich alle Achtung an Roelof Koning!
Ich habe eine 21 MB Festplatte und mir auch alle Komponenten gekauft, um diese IDE Schnittstelle für den SAM (!) nachzubauen (Erfahrungsbericht folgt). Unglücklicherweise ist die Hardware das einfachste Teil vom ganzen, man braucht aber ein gutes DOS (Disk operating system), um sie sinnvoll nutzen zu können. Die meisten Floppy Systeme, einschließlich 'MasterDos' beim SAM sind für eine Umprogrammierung auf Festplatte nur bedingt geeignet. Wer Lust hat, ein existierendes DOS zu modifizieren oder ein neues DOS zu schreiben, braucht eine genaue Beschreibung der IDE Schnittstelle. Den Anfang hat Roelof in seinem Beitrag gemacht und ich will versuchen dies hier zu erweitern und hoffe, das ein paar Leute ebenfalls Lust wie ich haben, bei diesem Thema einzusteigen.
Als erstes braucht man eine geeignete Festplatte, es gibt im Prinzip 3 Typen: die alten MFM Platten, die zwei Datenstecker hatten (sind ausgestorben), die 40 Pin IDE Schnittstelle, besser bekannt als AT-Bus Festplatten und die neuen SCSI (ausgesprochen scazi) Geräte. Nur eine AT-Bus Festplatte kann hier verwendet werden, diese sind jedoch sehr preiswert und werden von jedem Händler in Größen von 100 bis 500 MB angeboten. Auf Flohmärkten gibt es oft kleinere Platten von 21 bis 80 MB günstig zu erwerben. Sie haben alle hinten einen 40 Pin IDE Stecker und einen 4 PIN Netzstecker für +5 und +12 Volt.
Es ist in Wirklichkeit viel einfacher eine AT Platte zu programmieren als ein Floppy Laufwerk. Beim Floppy muß jedes Byte im genau richtigen Moment vom Rechner geschickt oder abgelesen werden (beim SAM muß man sehr geschickt programmieren, um das zu schaffen). Bei der AT Festplatte ist das kein Problem, sie hat eine 'eigene Intelligenz' eingebaut und kann von alleine Sektoren (512 Bytes) von der Festplatte lesen oder schreiben. Außerdem hat sie einen eigenen Puffer für die ganzen Daten, sodaß unser Rechner diesen Puffer in aller Ruhe laden oder ablesen kann. Alles andere macht die Platten-Elektronik selbst.
Einige Commands und ein bißchen Information braucht die Platte trotzdem, bevor sie gelesen oder beschrieben werden kann. Ein Beispiel dafür hat Roelof in seinem Beitrag gezeigt. Warum jedoch ist z.B. 33 ein 'Read Sector Command'? Schauen wir uns dies in binär an, damit es klarer wird:

COMMAND Bit:

7 (128	6 64	5 32	4 16	3 8	2 4	1 2	0 1)	
0	0	0	1	x	x	x	x	Recalibrate
0	0	1	0	0	0	L	R	Read Sector
0	0	1	1	0	0	L	R	Write Sector
0	1	0	0	0	0	0	R	Read Verify
0	1	0	1	0	0	0	0	Format Track
0	1	1	1	x	x	x	x	Seek
1	0	0	1	0	0	0	0	Execute Diagnostics
1	0	0	1	0	0	0	1	Set Parameters
1	1	0	0	0	1	0	0	Read Multiple
1	1	0	0	0	1	0	1	Write Multiple
1	1	0	0	0	1	1	0	Set Multiple Mode
1	1	1	0	0	1	0	0	Read Sect.Buffer
1	1	1	0	1	0	0	0	Write Sect.Buffer
1	1	1	0	1	1	0	0	Identify Drive
1	1	1	0	1	1	1	1	Set Buffer Mode
1	1	1	0	P	P	P	P	Power Commands

L=Data Mode select Bit:
0=Read/Write Data only
1=Read/Write Data and ECC
R=Retry Mode select Bit:
0=Retry enabled
1=Retry disabled
P=Valid bit for power commands
X=Unused bits, normally '0'

Es gibt viele Commands. Das Command 33 ist z.B. Bit 5 und Bit 0, beide sind auf eins gesetzt, was bedeutet: 'Read Sector mit Retry disabled'. Dies bewirkt, das ein Sector gelesen wird, wenn er aber fehlerhaft ist, wird kein zweiter Versuch unternommen. Mit Command 32 hätten wir das gleiche, aber bei einem Fehler würde die Platte automatisch mehrmals versuchen die Daten zu lesen, bevor sie aufgibt. Es ist auch möglich, Daten mit ECC zu schreiben und zu lesen. Der EEC ist ein 'Error Correcting Code', eine Art 'Prüfcharakter', der der Disk Elektronik erlaubt, bestimmte Lesefehler automatisch zu korrigieren ohne nochmals zu lesen. Normalerweise können bis zu 256 Sektoren von je 512 Bytes gelesen oder beschrieben werden, abhängig vom Inhalt des 'Sector-Counters'. Nach jedem Sektor wird ein Interrupt generiert, der für das Spectrum Interface unwichtig ist, weil er nicht benutzt wird. Es gibt aber ein Command 'Read/Write

Multiple'. Um diesen zu verwenden muß erst der 'Sector-Counter' mit einem Wert geladen werden, z.B. 4. Nun kann man ein 'Set Multiple command' durchführen. Es werden nun bei jedem 'Read/Write Multiple' command immer 4 Sektoren gelesen oder beschrieben und dann ein Interrupt generiert. Das ist sehr nützlich, weil ein PC immer Daten in Clusters liest oder schreibt. Ein typischer Cluster ist 2048 oder 4096 Bytes groß. Für das Spectrum HD ist dieser Command nicht so wichtig.

Interessant ist der 'Identify Drive' command. Mit diesem können wir Informationen von einem unbekannten (z.B. von einem auf einem Flohmarkt gekauften Gerät) Drive 'identifizieren'. Nach Ausführung des Command können wir 512 Bytes vom Puffer lesen, die so aussehen:

00/01	Konstant (0A5AH)
02/03	Number of Cylinders (Tracks)
04/05	Number of removable Cylinders (0000H)
06/07	Number of Heads
08/09	Number of bytes/track (unformated)
10/11	Number of bytes/sector (unformated)
12/13	Number of sectors/track
14/15	Number of bytes in the inter-sector gaps
16/17	Number of bytes in the sync fields
18/19	Reserved (000H)
20-39	Serial number of the disk
40/41	Controller type ex. (0003H)
42/43	Controller buffer size in 512 byte increments
44/45	Number of EEC bytes passed for R/W long commands
46-53	Controller firmware revision
54-93	Model Number
94/95	Number of sectors/interrupt (normally 0000H)
96/97	Double word transfer flag (normally 0000H)
98-511	Reserved (0000H)

Mit diesen Informationen wissen wir, wieviele Spuren unsere Platte hat, wieviele Schreib-/Leseköpfe usw. Die Anzahl von Bytes/Track und Sektor (unformated) ist sehr nützlich für den 'FORMAT' Command, der selten verwendet wird, etwas überraschend für einen Disketten-Benutzer. Wir sind gewöhnt, neue Disketten einfach zu formatieren, und auch um alte Disketten zu löschen greift man schnell zum 'FORMAT' Command, weil er alles löscht. Der Format Command macht aber eine ganze Menge mehr als nur die Daten zu löschen, er schreibt Informationen auf die Platte, die die Anzahl der Sektoren spezifizieren plus zusätzliche Synchronisations-Informationen, die notwendig sind um das Lesen und Schreiben von Daten zu erlauben. Eine Festplatte wird normalerweise nur einmal richtig formatiert, und das wurde bereits gemacht, bevor wir die Platte kauften. Dieses Vorformatieren wird 'Hard format' genannt, und was der Benutzer später 'Format' nennt ist nichts anderes, als ein Schreiben von '0000' oder ähnlichem in jeden Sektor oder mindestens das Überschreiben der Filenamen, sodaß ein 'DIR' oder 'CAT' Befehl die Platte leer erscheinen läßt. Es gibt verschiedene Gründe, warum eine Festplatte meistens nur einmal formatiert wird, z.B. dauert es sehr viel länger als bei einer einfachen Diskette (500 MB statt 780 KB). Ist bei einer Diskette ein Sektor defekt, schmeissen wir sie einfach weg, bei einer Festplatte würde das ein teurer Spaß, und fast JEDE Festplatte hat defekte Sektoren, die aber während des Formatierens durch eine Sonderprozedur entdeckt und 'ausgeklammert' werden, sodaß sie später nicht sichtbar sind.

Ein kleiner Fehler ist in den Originalbeitrag von Roelof eingeschlichen: die TASK Adresse zum STATUS lesen ist 247 (nicht 248).

Wenn Interesse besteht, dann bin ich gerne bereit, mehr zu diesem Thema zu schreiben. Aber ich glaube, das sich die Anzahl der Spectrum/SAM Benutzer mit Festplatten in Grenzen halten wird. Über meine Erfahrung mit der IDE Schnittstelle am SAM werde ich berichten, wenn meine Experimente abgeschlossen sind.

Ian D. Spencer, Fichtenweg 10c
53804 Much, Tel. 02245/1657

SAM Tips

Ich glaube, das manche gerne mal ein Programm auf dem SAM schreiben würden, ihnen aber gewisse 'liebe' Kenntnisse, die sie vom Spectrum her kennen, fehlen. Deshalb hier zwei Tips, die vielleicht noch nicht jeder SAM-User kennt.

Eingabe in Großbuchstaben (CAPS ON)

Was beim Spectrum der Befehl POKE 23658,8 bewirkte, können wir auch beim SAM einsetzen. Hier gilt: **POKE SVAR 618,8**.

Zeiterfassung mit dem SAM

Für bestimmte Spiele, wie etwa Reaktionstest, ist eine Routine nützlich, die für uns die benötigte Zeit festhält. Probiert mal folgendes:

```
10 POKE SVAR 632,0,0,0: REM Uhr reset
20 DEF FN zeit()=(65536*PEEK SVAR 634 +
   256*PEEK SVAR 633 + PEEK SVAR 632)/
   50: REM Funktion zur Zeitmessung
   einsetzen
30 LET uhr=FN zeit(), minute=INT (uhr
   /60),sekunde=INT (uhr-minute*60):
   REM Funktion zur Berechnung nutzen
40 PRINT AT 0,0;"Minuten: ";minute;" "
   ''"Sekunden: ";sekunde;" "
50 GO TO 30
```

(WoMo-Team)

Crack YS Games (2)

Also, nachdem die Hitze des Sommers weg ist, bekommt man wieder Lust aufs Cracken. Diesesmal nehmen wir außer YS auch SU Programme auseinander. Am besten beginne ich dort, wo ich im Teil 1 aufgehört habe. Also, wenn man die Befehlsfolge LD A,# oder irgendeine andere Operation am Register A, z.B. OR A, dann SCF gefunden hat, ersetzt man die weiteren Bytes durch CALL 1366, RET. Durch diese kleine Änderung wird durch das Programm statt der Speedloader Routine die im ROM vorhandene Load Routine aufgerufen. Der SCF Befehl muß so belassen werden, wie er ist, er sorgt dafür, das geladen und nicht veryfiziert wird. In das A-Register kann man aber jeden beliebigen Wert einsetzen, üblich ist der Wert 255, der dem normalen BYTES-Header entspricht.
Ich habe jedenfalls schon ein kleines Programm geschrieben, das den Loader von SU Games automatisch so umschreibt, das der Code mit normaler Geschwindigkeit geladen wird.
Der nächste Punkt sind die Programme, in deren Loadern scheinbar nichts sinnvolles ist, also wenn man kein LD IX,Nummer findet. In diesem Fall kann es sein, das der Loader kodiert ist. Um es zu cracken, muß zuerst die Kodierung weg. Die Dekodierungsroutine findet man an der Einsprungsadresse. An deren Ende setzt man nun EI, RET und startet sie mit RAND USR Einsprungsadresse. Nach ein paar Sekundenbruchteilen meldet sich der Computer mit der OK-Meldung zurück und man kann im dekodierten MC Listing lesen, wie die Startadresse u.s.w. sind. Natürlich sollte man darauf achten, das der RET an der richtigen Stelle sitzt, sonst hängt sich der Rechner auf! Wer ein Beispiel will, kann sich den Loader von EXPLORER ansehen.
Spiele, die ein eigenes Aufzeichnungsformat benutzen, können manchmal auch ganz einfach überlistet werden, ohne sich dabei viel Mühe zu machen. Man sollte wie immer die Startadresse, Länge und Aufrufadresse herausfinden, und dann zu dem Loader nur noch ein paar Befehle dazuschreiben, die mit Hilfe der ROM-Routinen dann den Code abspeichern.
Lange Files können auch gecrackt werden, indem man sie mit Hilfe von COPY COPY in drei Teile spaltet. Teil 1 ist der Screen, Teil 2 ist der Code von 23296 mit der Länge von 1704 Bytes und Teil 3 von 25000 bis zum Ende. Den Code von 23296 erweitert man mit einer MC Routine, die den Code von 16384 mit Hilfe von LDIR auf die Addresse 23296 kopiert und dann den eigentlichen Code mit JP oder CALL startet. Diesen Teil muß man dann auf die Adresse 16384 laden und die Routine aufrufen, wenn das Spiel gestartet werden soll.
Man sollte nicht vergessen LD SP,# durchzuführen, es ist so etwas wie CLEAR Addresse in BASIC.

Wer noch Fragen hat, verschiedene Crackertools haben will, oder nur ganz einfach Kontakt aufnehmen will, kann sich an die untenstehende Addresse wenden.

L.C.D., Zieglergasse 98/9
A-1070 Wien, Österreich

Plus D Tasword 128 Fix

Miles Kinloch hat einige 'Bugs' in der Tasword 128K Plus D Version gefunden und beseitigt. In seiner allerneuesten Version fügte er noch einige Pokes an, die nun auch verhindern, das aus dem Hauptmenu nach Wechseln auf das zweite Laufwerk, das Programm wieder zum ersten wechselt und dort versucht, ein File zu erasen.

```
 10 REM TASFIX by Miles Kinloch
 20 REM (PD) 1994
 30 REM
 40 REM FOR +D CONVERTED VERSION OF
    TASWORD 128. FIXES DATA MERGE AND
    CORRECTS BUG IN ERASE FILE OPTION.
    ALSO IMPROVES ERROR HANDLING.
 50 REM
 70 CLEAR 25269: PRINT "Insert the
    Tasword disk in Drive 1, then
    press any key."
 80 LOAD d1"tascode"CODE
 90 FOR a=30738 TO 30747: READ d: POKE
    a,d: NEXT a
100 DATA 205,243,120,56,5,202,207,119,
    24,54
110 FOR a=30790 TO 30797: POKE a,0:
    NEXT a
120 POKE 30763,0: POKE 30764,0: POKE
    30765,0: POKE 27667,27: POKE 27668,
    45
130 FOR a=52382 TO 52390: READ d: POKE
    a,d: NEXT a: DATA 33,155,204,23,
    123,204,205,212,98
140 POKE 61088,195: POKE 61089,152:
    POKE 61090,241
150 FOR a=61848 TO 61856: READ d: POKE
    a,d: NEXT a: DATA 58,210,202,50,89,
    240,207,65,201
160 CAT *"tascode": INPUT "Tascode
    start address"'"(25270 or 25300)? "
    ;start
170 PRINT ''"Any key to resave Tasword
    code.": PAUSE 0
180 SAVE d*"tascode" CODE start,65536-
    start
190 CLS: PRINT "Finally, add the
line:-"
    ''TAB 11;"5 GO TO 10"''"to the
    Tasword BASIC. This will cure the
    problem of the program locking up
    if you try to save to a write-
    protected disk."
```

Miles Kinloch, Flat 16, 6 Drummond Street
Edinburgh EH8 9TU, Scotland, U.K.



FREI BELEGBARE FUNKTIONSTASTEN!

Wer schon mal an einem PC gesessen hat (fast alle außer vielleicht W.H.), kennt die oberste Tastenreihe der Funktionstasten. Auf dem Spectrum gibt es sie leider nicht.

Dabei wäre es doch soooo praktisch, die unangenehme Syntax der diversen Datenspeicher mit einem Tastendruck zu erzeugen.

Aber es gibt ja meine MC-Hexenküche (in der ich alles sammle) und R.W.Gerling (der das mal ausgeheckt hat), und somit auch am Speccy Funktionstasten! Also gleich mal das Listing reinhacken, danach erkläre ich, wie's geht:

```
10 CLEAR 65022: FOR i = 65030 TO 65190: READ a : POKE i,a : NEXT i
20 DATA 62,62,237,71,237,86,201,59,93,0,62,253,237,71,33,255,253,54,42,35,54,254,
237,94,42,83,92,34,13,254,62,0,50,15,254,201,255,243,221,229,229,213,245,58,15,
254,254,1,40,59,253,203,1,110,40,97,205,30,254,221,42,13,254,221,126,4,254,234,
32,80,221,126,5,33,8,92,190,40,20,221,110,2,221,102,3,237,91,13,254,25,17,4,0,
25,34,13,254,24,216,205,137,254,62,1,50,15,254,24,44,221,42,13,254,221,126,7,254,
13,40,30,254,64,32,2,62,13,205,140,254,24,22,221,126,7,50,8,92,33,59,92,203,238,
42,13,254,35,34,13,254,201,205,30,254,241,209,225,221,225,251,201,0
```

Dann wird die mühevolle Tipperei mit CODE 65030,160 auf Disk oder Tape gerettet.

Und so funktioniert es: Zuerst wird das Programm initialisiert. Dies geschieht mit RANDOMIZE USR 65040.(Deaktiviert wird es mit USR 65030). Jetzt könne wir die Tasten definieren. Sagen wir zB wir wollen die Taste % mit der Funktionenfolge BORDER 0: PAPER 0: INK 7: CLS belegen. Praktisch wenn die Farben verstellt sind oder zum neu Beginnen. Zu diesem Zweck schreiben wir in eine Basiczeile folgendes:

```
10 REM %: BORDER 0: PAPER 0: INK 7: CLS @
```

Das letzte Zeichen (Klammeraffe) wird mit Symbol Shift und 2 erzeugt und bewirkt, daß die Zeile sofort ausgeführt wird.
Will man nochwas anhängen, zB beim Laden von BETADISK den Filenamen, läßt man dieses Abschlußzeichen weg:

```
10 REM %: RANDOMIZE USR 15363:REM LOAD"
```

Die Zeile erscheint dann am Screen und kann bearbeitet werden.
(ACHTUNG: Beispielzeile 2 mit dem USR-Aufruf geht natürlich nur für Betadisk!!!!)

Natürlich können mehrere solcher Zeilen definiert werden, jeweils eine Taste pro Zeile.

Noch einige Hinweise: Um eine klaglose Funktion zu gewährleisten, sollten die Definitionen am Anfang des Programmes stehen, der Tastenpieps mit POKE 23609,0 abgeschaltet werden und die Befehlszeilen möglichst kurz sein. Außerdem sollte die betreffende Taste mindestens 1 Sekunde gehalten werden.

Soviel zu Teil 3. Auch in Teil 4 bekommt der Speccy wieder etwas aus der PC-Welt: Es geht um Bildschirmschoner....

SpecTec, Christoph Kment, Trabertgasse 42, A-1130 WIEN

Heh! Wer ist denn da mit W.H. gemeint? Wenn es der ist, den ich meine: Gerade weil er schon am PC gesessen hat, liebt er diese so... (W.H.)



Kalender oder auch "Kalendermacher" genannt, ist eines der besten Programme die von Walter Sperl für DTP geschrieben wurden. Wer also dieses Programm auf Kassette ruhen läßt ist wirklich selber Schuld. Ihr findet dieses Programm auf dem Datenpack II von Herbert Hartig aus Buchloe. Kalender besteht aus 8 selbständigen Basic-mc-Programmen die über ein Menue (**Bild 1**) verwaltet werden. Diese Kalender - Programme generieren DTP - Textfiles zur **individuellen Gestaltung** von Kalendern mit dem DTP-Typeliner. Zusätzlich wurde von W. Sperl das **Anwenderpaket "kalpge"** angelegt. Es enthält neben einigen Kalenderschriften (Font 3 - Zeichensätze) auch mehrere Demo - Pages. Außerdem ist das Einbinden von Kalenderteilen in die Textverarbeitung in Anlehnung an den TEXT-O-mat (Herbert Hartig) möglich. Es empfielt sich als erstes alle Programmteile (+ "run"-File) auf Diskette oder Microdrive-Cartrige zu kopieren.

Jahr (1901-2099) 1995L

Start: Kalender mit Hilfe des "Run-Files" laden. Es erscheint das Verwaltungsmenue (**Bild 1**) von Kalender. Ihr könnt jetzt das gewünschte Programm mit den **Tasten 1 - 8** anwählen. Als nächstes wird das gewünschte Jahr eingegeben + Enter. Ihr könnt zwischen (**BILD 2**) dem **Jahr 1901** bis **2099** wählen. Im "Normalfall" dürfte diese Spanne ja wohl ausreichen, oder? Kalender braucht jetzt ein paar Sekunden (**Bild 3**) um das gewählte Jahr zu generieren. Es folgt die Meldung **"Start tape. then press any key"** , zusätzlich

KALENDER

ohne Feiertage	mit Feiertagen
1 Wochenbeginn Montag	4 Wochenbeginn Montag
2 Wochenbeginn Sonntag	5 Wochenbeginn Sonntag
3 Wochentage vertikal	6 Wochentage vertikal
	7 Monatskalender
	8 Wochenkalender

Bitte eine Zahl tippen

Ich arbeite
und Du schaust zu!

wird die Filelänge (**Bild 4**) angezeigt. Nach dem Saven könnt Ihr sofort ein weiters Jahr bearbeitet. Wird die Frage (**Bild 5**) mit **"N"** beantwortet geht es zurück zum Menue. Ihr könnt jetzt das "Kalender-File" in den Wordmaster laden und mit dem Typeliner (+ kalpge) weiter arbeiten. • **ENDE** •

Ich arbeite
und Du schaust zu!

File-Laenge = 3240

Start tape, then press any key.

In Teil 33 geht es dann um den **"HEADLINER !"** und seinen Zusatzroutinen. Bis dahin wünsche ich Euch noch viel Spaß mit DTP LG 32.•

Ich arbeite
und Du schaust zu!

File-Laenge = 2471

Noch ein Jahr (j/n)?

Oldenburg, den 03. 05. 1994

SPECTRUM KALENDER 1995

So	1	Jänner 1995
Mo	2	
Di	3	BEGINN: SKIURLAUB
Mi	4	
Do	5	
Fr	6	
Sa	7	
So	8	
Mo	9	
Di	10	
Mi	11	
Do	12	
Fr	13	
Sa	14	
So	15	
Mo	16	
Di	17	
Mi	18	GEBURTSTAG: GABY
Do	19	
Fr	20	
Sa	21	
So	22	
Mo	23	
Di	24	URLAUB: ENDE
Mi	25	
Do	26	
Fr	27	
Sa	28	
So	29	
Mo	30	
Di	31	

Februar 1995					
So		5	12	19	26
Mo		6	13	20	27
Di		7	14	21	28
Mi	1	8	15	22	
Do	2	9	16	23	
Fr	3	10	17	24	
Sa	4	11	18	25	

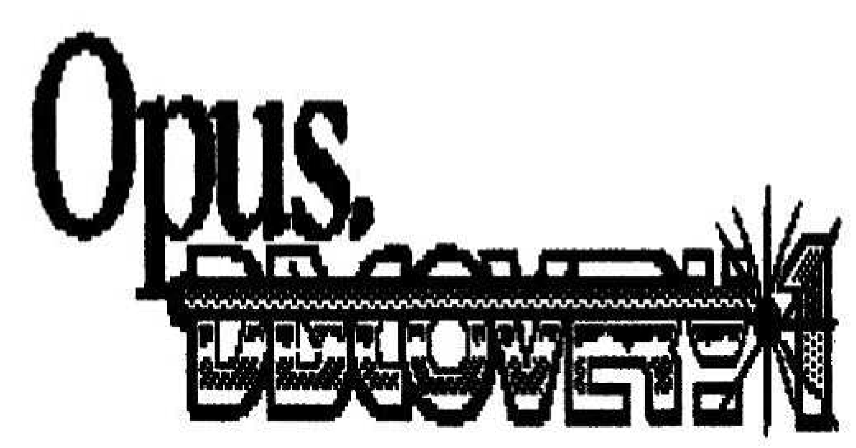
März 1995					
So		5	12	19	26
Mo		6	13	20	27
Di		7	14	21	28
Mi	1	8	15	22	29
Do	2	9	16	23	30
Fr	3	10	17	24	31
Sa	4	11	18	25	

April 1995						
So		2	9	16	23	30
Mo		3	10	17	24	
Di		4	11	18	25	
Mi		5	12	19	26	
Do		6	13	20	27	
Fr		7	14	21	28	
Sa	1	8	15	22	29	

Mai 1995					
So		7	14	21	28
Mo	1	8	15	22	29
Di	2	9	16	23	30
Mi	3	10	17	24	31
Do	4	11	18	25	
Fr	5	12	19	26	
Sa	6	13	20	27	

Juni 1995					
So		4	11	18	25
Mo		5	12	19	26
Di		6	13	20	27
Mi		7	14	21	28
Do	1	8	15	22	29
Fr	2	9	16	23	30
Sa	3	10	17	24	

Juli 1995						
So		2	9	16	23	30
Mo		3	10	17	24	31
Di		4	11	18	25	
Mi		5	12	19	26	
Do		6	13	20	27	
Fr		7	14	21	28	
Sa	1	8	15	22	29	



Direkte Ansteuerung der Drives

Diesen Monat zuerst noch eine kleine Ergänzung zum Artikel aus dem letzten Info. Die MC-Routine des OPUS-Info-Sektors enthält zu Beginn fünf Datenbytes, wobei ich bei den letzten beiden Bytes geschrieben habe, sie hätten keine Verwendung. Dies ist nicht richtig. Bevor der Info-Sektor geschrieben wird liest der Spectrum den Inhalt der Speicherstellen 23672 und 23673 aus und schreibt diese zwei Bytes in den Infosektor. Die genannten Speicherstellen gehören zur Spectrumuhr, welche 50-stel Sekunden zählt, was bedeutet, daß die zwei Speicherstellen 50 mal in der Sekunde ihren Wert ändern (nicht, wenn der Interrupt ausgeschaltet ist). So erhält jede Diskette eine "einzigartige" Kennnummer. Kommen wir zum heutigen Thema.

Natürlich gibt es im OPUS-ROM Routinen, welche einzelne Sektoren lesen oder schreiben, aber wenn wir diese benützen, lernen wir nichts dabei und außerdem sind die selbstgeschriebenen Routinen viel schneller. Wie im August kurz erwähnt, dienen die Adressen 10240-10243 zur Steuerung der Drives. Auch brauchen wir die Adresse 12288. Hier eine kleine Tabelle, was die einzelnen Adressen bedeuten:

Adresse	Lesen	Schreiben
10240	Statusregister (Bitnr.):	Kommando an Diskcontroller
	0: 1: Kommando wird gerade ausgeführt	Name (Typ) — Bits 76543210
	0: Warten auf neues Kommando	Restore Drive (I) 0000HVSR (fährt Arm auf Track 0)
	1: 1: Zeigt an, daß das Datenregister einen Wert enthält (lesen) bzw leer ist, wenn eine Schreib-operation ausgeführt wird	Step-In Arm (I) 010UHVSR (bewegt Arm einen Track nach innen) Step-Out Arm (I) 011UHVSR (bewegt Arm einen Track nach außen)
	2: 1: Der Computer hat kein Datum in der nötigen Zeit gesandt. Bei Kommandotyp I: Der Zustand des Track 0-Signals	Suche Track (I) 0001HVSR (vorher in Adresse 10243 die gewünschte Tracknummer schreiben, der Arm wird dorthin bewegt)
	3: 1: Fehler im Datenfeld bzw. ID-Feld eines Sektors	Lese Sektor (II) 100MHE00 (liest den durch Adresse 10242 bestimmten Sektor auf dem Track, der sich gerade unter dem Arm befindet)
	4: 1: Der angesteuerte Track, Sektor oder Seite konnte nicht gefunden werden	Schreibe Sektor(II) 101MHEPA (schreibt den durch Adresse 10242 bestimmten Sektor auf den Track, der sich gerade unter dem Arm befindet)
	5: Kommandotyp I: 1: Die Motoranlaufphase ist beendet (6 Umd.) Kommandotyp II, III: 1: Gelöschte Datenmark. 0: Normale Datenmark.	Lese Trackinfo (III) 110MHE00 (liest die als nächstes zur Verfügung stehenden Sektormarkierungen)
	6: 1: Disk schreibgeschützt	Schreibe Track (III) 1111HEP0 (formatiert Track bzw. Sektor)
	7: 1: Motor ist ein 0: Motor ist aus	Erzwinge Interrupt 1101IJ00 (IV) (??)

10241 Trackregister (enthält die aktuelle Tracknummer)

10242 Sektorenregister (enthält die aktuelle Sektorennummer)

10243 Datenregister (zum Lesen und Schreiben von Diskdaten)

12288 Bit 0: 0: Drive 1 unbenutzt
1: Drive 1 benutzt
Bit 1: 0: Drive 2 unbenutzt
1: Drive 2 benutzt
Bit 4: 0: Diskseite 1 aktiv
1: Seite 2 aktiv
Bit 5: 0: DD-Laufwerk
1: SD-Laufwerk
(die anderen Bits werden nicht zur Laufwerkssteuerung benötigt)

Die einzelnen Flags der Kommandos bedeuten dabei:

A: 0: Es wird eine normale Datenmarkierung geschrieben
1: Es wird keine Datenmarkierung geschrieben, was bei Leseversuchen mit BASIC-Befehlen zu Fehlermeldung führt
E: 1: 30 Millisekunden Wartezeit vor Ausführung
0: Keine Wartezeit
H: 0: Die Motoranlaufphase soll ausgeführt werden, falls der Motor noch nicht läuft
1: Auf keinen Fall eine Motoranlaufphase
I: Interruptzustand bei Indexpulsierung (??)
J: Beenden ohne weiteren Interrupt (?)
M: 0: Einfacher Sektor (?)
1: Mehrfacher Sektor (?)
P: 0: Schreibvorkompensierung erlaubt
1: nicht erlaubt
S,R: Gibt die Steppingrate an, also die Zeit, die das Laufwerk brauchen darf, um den Arm einen Track weiter zu bewegen
00:6ms 01:12ms 10:20ms 11:30ms
U: 0: Das Trackregister 10241 wird bei Step-In und Step-Out-Befehlen nicht verändert
1: Es wird entsprechend verändert
V: 1: Es wird überprüft, ob der gewünschte Track (10241) gefunden wurde
0: Keine Überprüfung (ist schneller)

Dies sind viele Informationen, von denen auch ich noch nicht alle verstanden habe (siehe Fragezeichen). Aber die meisten Dinge sind mir klar und so können wir jetzt unsere eigenen Routinen schreiben. Dazu eine wichtige Information: Wird bei dem Kommando, welches in Adresse 10240 geschrieben wird, das Bit 7 auf 1 gesetzt (also Kommandos des Typs II-IV), erzeugt die OPUS NMIs (Not Maskable Interrupts). Bei jedem dieser Interrupts muß der Computer an Adresse 102 des ROMs weiterarbeiten. Im OPUS-ROM steht hier der Befehl JP (HL). Es wird also zu einer Routinegesprungen, deren Adresse im HL-Register steht. Dies ist der wichtigste Punkt beim Entwickeln eigener Routinen und auch der Grund, warum die NMI-Leitung nicht an die rechte Seite der OPUS durchgeführt wurde. Es ist aber kein Problem, die Leitung selbst mit einem Kabel an den PTC zu führen, nur darf z.B. nie während einer Diskoperation das Multiface gedrückt werden.

Zum besseren Verständnis hier nun einige Assemblerroutinen:

```
ST_COM  EQU   10240        ;Status-/
                           ;Kommandoreg.
TRACK   EQU   10241        ;Tracknr.register
SEKTOR  EQU   10242        ;Sektornr.register
DATA    EQU   10243        ;Datenregister
FLAGS   EQU   12288        ;Flagregister

BUFFER  EQU   50000        ;Buffer im Speicher

        ORG   60000        ;MC ab Adr.60000

        CALL  5896         ;Opus einblenden
        DI                 ;kein Interrupt bei
                           ;Diskop.
        LD    A,(FLAGS)    ;Es wird das Lauf-
        AND   252          ;werk 1 ausgewählt
        OR    1
        LD    (FLAGS),A

        XOR   A            ;=> A=0
        CALL  WARTE        ;Laufwerk Reset

        LD    A,18         ;Arm auf Track 18
        LD    (DATA),A     ;fahren
        LD    A,%00010000  ;dabei Motoranlauf-
                           ;phase erlauben
                           ;keine Überprüfung,
                           ;ob richtiger Track
                           ;gefunden und 6 ms
                           ;Stepping
        CALL  WARTE

        LD    HL,LESEN     ;NMI-Routinen-
                           ;adresse
        LD    DE,BUFFER    ;hierhin sollen die
                           ;Diskwerte gelesen
                           ;werden
        LD    BC,256       ;es sollen 256
                           ;Bytes gelesen
                           ;werden
```

```
        XOR   A              ;A=0,SEK I.0 soll
        LD    (SEKTOR),A     ;gelesen werden
        LD    A,%10000000    ;Sektor lesen, dabei
                             ;einfacher Sektor,
                             ;Motor anlaufen er-
                             ;lauben, keine
                             ;Wartezeit vor
                             ;Ausführung
        CALL  WARTE
        EI                   ;Interrupts wieder
                             ;ein
        JP    5960           ;Opus aus und
                             ;BASIC

WARTE   LD    (ST_COM),A     ;Befehl an
                             ;Controller
        XOR   A
W1      DEC   A              ;eine kleine
                             ;Wartezeit
        JR    NZ,W1
W2      LD    A,(ST_COM)     ;Ist die Opus schon
        BIT   0,A            ;fertig?
        JR    NZ,W2          ;Nein, dann warten
        RET

LESEN   EX    AF,AF'         ;Register A sichern
        LD    A,B            ;sind schon alle
                             ;Bytes
        OR    C              ;gelesen (BC=0) ?
        LD    A,(DATA)       ;Diskdata lesen
        JR    Z,L1
        LD    (DE),A         ;in Speicher
                             ;schreiben
        INC   DE             ;naechste Adresse
        DEC   BC             ;ein Byte gelesen
L1      EX    AF,AF'         ;Register A
                             ;zurück
        RETN                 ;NMI-Routine
                             ;verlassen
```

Noch eine Anmerkung zum Suche Track-Befehl: Der Controller geht davon aus, daß der Wert des Trackregisters 10241 richtig ist. Steht der Arm z.B. über Track 7 und das Trackregister enthält nicht 7, sondern 11, und Sie wollen Track 14 ansteuern, so wird der Arm über Track 10 gefahren, da das Laufwerk annimmt, es müßte nur 3 Tracks (14-11=3) nach innen fahren. Haben Sie das V-Flag (Bit 2) auf 1 gesetzt, so erhalten sie in diesem Fall ein gesetztes Bit 4 im Statusregister (10240), welches anzeigt, daß der gewünschte Track nicht gefunden wurde. Wurde das V-Flag nicht gesetzt, so nimmt der Controller an, der Arm ist über dem richtigen Track, obwohl dies in Wahrheit nicht der Fall ist.

Falls mir jemand die unklaren Dinge (Fragezeichen) erklären kann, würde mich dies natürlich besonders freuen. Ansonsten verbleibe ich bis zum nächsten Monat.

Helge Keller, Hermann-Löns-Weg 51, 76307 Karlsbad, Tel.:07202/6076

Das BETA-Disk-IF

Heute nun der dritte Teil der Serie für die Beta Disk User, die in leicht veränderter Form von Wilhelm Wiecke im ehemaligen SUCW veröffentlicht wurde.

Heute fahren wir mit der Beschreibung der BETA-Disk Version 5.03 fort, diesmal geht es hauptsächlich um den vom TRDOS benötigten Puffer für die zusätzlichen Systemvariablen. Dafür werden 112 Bytes im RAM von Adresse $5CB6-$5D25 (23734-23845) benutzt. Da ab $5D26 noch die verschobenen Kanalinformationen folgen, kann ein Basicprogramm nun bei Adresse 23867 ($5D3B) beginnen. Die neuen Systemvariablen werden beim ersten Einschalten, oder falls zwischenzeitlich gelöscht, auch durch RANDOMIZE USR 15616 (oder ...15619: REM:...) initialisiert!
Nach dem Einschalten arbeitet der Rechner zuerst im BETA-ROM einige Routinen ab, die den originalen ROM-Routinen stark ähneln. Wer ein Spectrum-ROM Listing hat, kann mal vergleichen (im TRDOS von Adresse 0-$00F9 findet man weitgehend im ROM bei Adresse 0-4 und $11CC-$1287 wieder!). Es wird im wesentlichen der Speicher gecheckt, die UDG's kopiert und Systemvariablen gesetzt!
Durch eine gegenüber dem Spectrum-ROM modifizierte Kopierroutine ($00A5-$00B2 im Beta-ROM) werden nun zuerst die normalen Kanalinformationen ins RAM ab $5CB6-$5CCA kopiert. Nachdem weitere normale Systemvariablen mit ihren Ausgangswerten besetzt wurden, erfolgt bei Adresse $0114 ein JP $3D31. Zum Verfolgen des weiteren Geschehens dient das folgende Assemblerlisting:

```
31F3   ORG   $31F3

31F3   LD    HL,($5C4F)   ;$5C4F=CHANS ->
                          ;Adresse von
                          ;Kanalinformation
                          ;ins HL-Register
```

Wenn Bereich noch nicht erweitert, steht hier die Adresse $5CB6 (23734), sonst gewöhnlich $5D26 (=23734+112=23846).

```
31F6   OR    A            ;Flags löschen!
31F7   LD    BC,$5D25     ;Endadresse des er-
                          ;weiterten Bereichs
                          ;ins BC-Reg. laden
31FA   SBC   HL,BC        ;Subtrahiere HL-BC
                          ;unter Berücksichtg.
                          ;des Übertrag-Flags!
```

Je nachdem, ob in HL $5CB6 oder $5D25 steht, wird bei der Subtraktion das C-(Übertrag)-Flag gesetzt ($5CB6-$5D25) oder nicht ($5D26-$5D25)

```
31FC  RET                ;Sprung zurück nach
                         ;$3D24
```

Der Bereich von $31FD-$3C00 im Eprom ist leer ($FF)

```
3C01  ORG   $3C01
3C01  JR    $3C06        ;Direktaufruf d. Ver-
                         ;sion 3.xx landet hier
3C03  DB    $FF
3C04  JR    $3C09        ;Aufruf aus Basic bei
                         ;Version 3.xx landet
                         ;dagegen hier
3C06  JP    $3D00        ;Sprung erfolgt nach
                         ;15616
3C09  JP    $3D03        ;Sprung erfolgt nach
                         ;15619
```

Der Bereich von $3C0C-$3CF9 im Eprom ist leer ($FF). In diesem Bereich landen Aufrufe der Beta-Version 4.xx!

```
3CFA  ORG   $3CFA
3CFA  JP    $20F1        ;Aufruf einer Routine
                         ;zum Testen auf IF1
3CFB  JP    $283C        ;Sucht eine im C-Reg.
                         ;stehende Funktions-
                         ;nr. bei MC-Aufruf
3D00  NOP
3D01  JR    $3D31        ;Hier landet RANDOM.
                         ;USR 15616!
3D03  NOP
3D04  JR    $3D1A        ;Hier landet RANDOM.
                         ;USR 15619!
3D06  NOP
3D07  JP    $25EF        ;Bedeutung ungeklärt!
3D0A  JP    $244A        ;Desgleichen!
3D0D  NOP
3D0E  JR    $3D0A
3D10  NOP
3D11  JR    $3CFA
3D13  NOP
3D14  JR    $3CFB        ;Hier erfolgt meist
                         ;Aufruf des TRDOS
                         ;aus MC-Prog.!
```

Es müssen neben Funktionnr. noch div. Parameter übergeben werden. Beispiele folgen später!

```
3D16  NOP
3D17  JP    $2F69        ;Alle Fehlermeldungen
                         ;laufen über diese
                         ;Routine ab $2F69
3D1A  CALL  $3D21        ;Hier zur Erweiterung
                         ;testen, nach RAND.
                         ;USR 15619
3D1D  PUSH  HL           ;HL-Reg. enthält hier
                         ;Ergebnis der Subtr.
                         ;von Adr. $31FA
3D1E  JP    $016C        ;Bei $016C weiter-
                         ;machen (nach Aufruf
                         ;aus Basic)
3D21  CALL  $31F3        ;Testen, ob Puffer auf
                         ;112 Bytes erweitert
                         ;ist
3D24  NOP
3D25  NOP
3D26  CALL  C,$3D4C      ;Wenn nicht erweitert
                         ;(C-Flag gesetzt!),
                         ;nach $3D4C
3D29  LD    HL,$5CC2     ;Sonst HL-Reg. mit
                         ;5CC2 laden!
```

Dient später als Rücksprungadresse und enthält einen RET-Befehl.

```
3D2C  RET                ;Zurück nach $3D1D
                         ;oder $3D34, je nach-
                         ;dem, woher Aufruf
                         ;kam.
3D2D  NOP
3D2E  NOP
3D2F  NOP
3D30  RET
3D31  CALL  $3D21        ;Testen, ob Erwei-
                         ;terung auf 112 Bytes
                         ;schon geschehen.
```

Hierher erfolgte der Sprung bei Adr. $0114 im Beispiel oder nach Aufruf mit RANDOMIZE USR 15616!

```
3D34  PUSH  HL           ;HL-Reg. enthält
                         ;$5CC2 als Rück-
                         ;sprungadr. (RET)
3D35  JP    $0239        ;Hier geht's anschlies-
                         ;send weiter!
```

Der folgende Bereich ($3D38-$3D4B) wird später in den MEMBOT-Bereich kopiert und gestartet. (Test ob RS-232 oder ZX-Printer angeschlossen)

```
3D38  XOR   A            ;A-Reg. löschen
3D39  OUT   ($F7),A
3D3B  IN    A,($F7)      ;Serielle Schnittstelle
                         ;(Port $F7) abfragen?
3D3D  CP    $1E          ;RS-232 Ausgabe
                         ;möglich?
3D3F  JR    Z,$3D44      ;Dann weiter bei
                         ;$3D44!
3D41  CP    $1F          ;ZX-Printer Ausgabe?
3D43  RET   NZ           ;Wenn nicht zurück!
3D44  RST   $08          ;Bei IF 1 wird dies
                         ;aufgerufen und der
                         ;nachfolgende
3D45  DB    $31          ;Befehl 'Variablen
                         ;initialisieren' aus-
                         ;geführt!
3D46  LD    A,$01        A-Reg. mit 1 laden
                         ;als 'Merker' für ini-
                         ;tialisierte RS-232
3D48  LD    ($5CEF),A    ;Schnittstelle und dies
                         ;in $5CEF retten!
3D4B  RET                ;Zurück!
3D4C  XOR   A            ;A-Reg. mit 0 laden.
3D4D  OUT   ($FF),A      ;???
3D4F  IN    A,($F6)      ;???
3D51  LD    HL,L3D38     ;Anfangsadr. des zu
                         ;kopierenden Bereichs
                         ;ins HL-Reg.
3D54  LD    DE,$5C92     ;Adresse von MEMBOT
                         ;als Zieladresse
                         ;ins DE-Register
```

```
3D57  LD    BC,$0014   ;und Zeichenanzahl in
                       ;BC laden!
3D5A  LDIR             ;Die 20 Zeichen ab
                       ;$3D38-$3D4B nach
                       ;$5C92 kopieren!
3D5C  LD    HL,$3D67   ;Adr. wird nach Aus-
                       ;führung des JP bei
                       ;$3D64 angesprungen!
3D5F  PUSH  HL         ;Dazu auf den Stapel
                       ;ablegen.
3D60  LD    HL,$3D2F   ;Diese Adr. enthält
                       ;einen RET-befehl.
3D63  PUSH  HL         ;Auch auf den Stapel
                       ;als Rücksprungadr.!
3D64  JP    $5C92      ;Aufruf der kopierten
                       ;Routine in MEMBOT
                       ;im RAM!
```

Nach Rückkehr aus der Routine gehts nach $3D2F, was ein RET bewirkt im Beta-ROM nach $3D67!

```
3D67  LD    HL,$2F90   ;HL mit Startadr. für
                       ;Routine 'Defaultwerte
                       ;setzen' lassen
3D6A  PUSH  HL         ;und auf den Stapel!
3D6B  LD    HL,$3D2F   ;Adr. zeigt auf RET-
                       ;Befehl.
3D6E  PUSH  HL         ;Auch auf den Stapel!
3D6F  LD    HL,$1655   ;ROM-Routinen Adr.
                       ;zum Reservieren von
                       ;Speicherplatz.
```

Bytezahl dazu in BC übergeben, HL zeigt auf Stelle, hinter der Platz geschaffen werden soll!

```
3D72  PUSH  HL         ;$1655 auf Stapel
                       ;ablegen.
3D73  LD    HL,$5BFF   ;Adr. des Drucker-
                       ;pufferendes (bei 48K
                       ;Spectrum) in HL!
3D76  PUSH  HL         ;Auch retten!
3D77  LD    (HL),$C9   ;In $5BFF (im RAM)
                       ;ein RET schreiben.
3D79  LD    HL,$5CB5   ;P-RAMT = letztes
                       ;Byte der normalen
                       ;Systemvariablen.
```

Dahinter werden durch ROM-Routine $1655 die 112 Bytes reserviert!

```
3D7C  LD    BC,$0070   ;BC-Reg. mit 112 (=
                       ;Byte-Anzahl) laden.
3D7F  RET              ;Zurück nach $5BFF
                       ;im RAM, was erneut
                       ;ein RET bewirkt.
```

Und nun gehts weiter im ROM bei $1655. Bei Rückkehr wird dann $3D2F im Einschaltbereich des Beta-ROM erreicht: ein erneutes RET ist die Folge nach $2F90 auch im Beta-ROM. Nach Ende wird nun $3D29 auf dem Stapel gefunden, damit ist die Pufferinitialisierung beendet. Dort weiter!

```
3D80  END
```

Jetzt noch die Routine (ab $2F90), die den geschaffenen Pufferbereich mit den wichtigsten Ausgangswerten füllt:

```
2F90  LD    HL,$FFFF
2F93  LD    ($5CFA),HL ;Die folgenden Adr.
                       ;mit $FF laden. Hier
2F96  LD    ($5CFC),HL ;stehen später Infor-
                       ;mationen zu den
2F99  LD    ($5CC8),HL ;Laufwerken.
2F9C  LD    ($5CCA),HL
2F9F  XOR   A          ;A-Reg. mit 0 laden.
2FA0  LD    ($5D17),A  ;$00 ganzen Screen,
                       ;$AA untere 2 Zeilen
                       ;löschen.
2FA3  LD    ($5D19),A  ;Laufwerk 0 (= A)
                       ;setzen. Möglich sind
                       ;Werte von 0-3!
2FA6  LD    ($5D18),A  ;$00 Kein IF 1, $FF
                       ;IF 1 angeschlossen!
2FA9  LD    ($5D0F),A  ;Fehlernummer
2FAC  LD    ($5D1F),A  ;???
2FAF  LD    A,$FF      ;A-Reg.m. $FF laden.
2FB1  OUT   ($FF),A    ;???
2FB3  LD    ($5C3A),A  ;ERR NR mit $FF für
                       ;Fehlermeldung 'OK'
                       ;laden!
2FB6  LD    ($5D16),A  ;Defaultwerte für
                       ;LW-Code setzen.
```

Hier stehen später für die Laufwerke A-D die Werte $3C-$3F, wenn Seite 0, und $2C-$2F, wenn Seite 1 der Diskette angesprochen wird!

```
2FB9  LD    ($5D0C),A  ;$00 Puffer-Erwei-
                       ;terung >$5D25 vor-
                       ;handen, $FF nein!
2FBC  LD    A,$C9      ;A-Reg. mit $C9 (=
                       ;RET) laden
2FBE  LD    ($5CC2),A  ;und an diese Adr.
                       ;schreiben (s. Bemerk.
                       ;bei $3D29)!
2FC1  LD    A,$D0      ;A-Reg. mit 208
                       ;laden.
2FC3  OUT   ($1F),A    ;???
2FC5  RET              ;Zurück nach $3D29,
                       ;dort weitermachen!
```

Auch wenn dies vielleicht nur für einige wenige User wirklich interessant war, geben die Routinen doch einen Einblick, wie trickreich und mit wie vielen Umwegen das Zusammenspiel zwischen BETA-ROM, SPECTRUM-ROM und RAM funktioniert, und das alles in Sekundenschnelle!
Vielleicht noch eine Erläuterung: Für das Verständnis der Routinen muß man wissen, das nur Aufrufe des Bereichs $385F (=14431) bis $3DFF (=15871) das BETA-ROM einschalten. Soll vom BETA-ROM das SPECTRUM-ROM aufgerufen werden, ist ein Umweg über das RAM notwendig! Das gleiche gilt natürlich auch im umgekehrten Fall. Besonders elegant geht das eben, wenn wie gesehen, die anzuspringenden Adressen auf dem Stapel deponiert werden und vorhandene RET-Befehle im jeweiligen Bereich zum Abheben der Adressen benutzt werden!
Für heute genug! Nächstes mal nähere Informationen dazu, welche Daten der Puffer im Einzelnen enthält und weitere Routinen. Bis dann!

***** Die Beta Connection *****
Jean Austermühle/Dirk Berghöfer

TIPS / TRICKS

Nochmals zum Beitrag "Monitoranschluß am Spectrum +2A"

Das über diesen Weg erzielte Bild ist schon eine wesentliche Verbesserung der Qualität. Leider störten bei 3 untersuchten Geräten noch Interferenzstreifen. Diese können in ihrer Stärke etwas verringert werden, wenn man den Emitter des zusätzlichen Transistors über einen Einstellregler von ca. 470 Ohm an Masse legt und den 75 Ohm Widerstand an den Schleifer legt. So lassen sich individuelle Einstellungen für das jeweilig benutzte Fernseh- bzw. Monitorgerät erreichen. Wie ich schon im Info 8/94 erwähnte, sollte eine Diode zwischen dem Encoder-Anschluß Pin 6 und der Basis eingefügt werden. Ebenso trägt ein Widerstand, ca. 6.8 KOhm, zwischen Basis des Transistors und Masse im Sperrbereich der Basis-Emitter-Diode des Transistors für stabilere Verhältnisse bei.
Eine absolute Beseitigung der Interferenzstreifen ist möglich (Bild fast ideal, wenn nicht noch einige geringe Farbverschiebungen übrig blieben), wenn man den Sound-Modulator-Oszillator totlegt. Dann ist im Gerät nur noch der Muttergenerator mit Quarz X1 (35,469 MHz) aktiv.
Der Sound-Modulator-Oszillator befindet sich unter den Joystick-Buchsen und dem Modulator VM 1233. Die ISS 1 benutzt eine Schaltung mit dem IC MC 1376, die ISS 4 eine Transistorschaltung. Die einfachste Methode ist nun, den Quarz X3 (6,0 MHz) mit einem 1uF-Keramik+Kondensator zu überbrücken.
Der VHF-Ausgang des +2A hat nun natürlich keine Sound-Modulation mehr. Es ist aber noch die Tape-/Sound-Ausgangsbuchse da, an die ein NF-Verstärker angeschlossen werden kann. Die Tonqualität ist dort ja sowieso auch besser. Wohlbemerkt, es ist dort ein Stereo-Klinkenstecker zu benutzen. An den zweiten Buchsenanschluß hat der Gerätehersteller einen Test-Eingang, der mit dem PLAY-Ausgang des eingebauten Kassettenrekorders verbunden ist (über Kondensatoren 1,0 uF) gelegt. Bei Verwendung eines Monosteckers würde der dann an Masse gelegt und es geht nichts mehr.
Dieser Testeingang ist prinzipiell auch als Eingang für ein anderes Kassetten-Datengerät verwendbar. Nur müßten da der PLAY-Ausgang des eingebauten Kassettenrekorders abgeschaltet werden, oder eleganter, Entkoppelungsmaßnahmen durchgeführt werden, z.B. am einfachsten durch 2 Dioden.
Meine Frage ist nun: Wer experimentiert da weiter?
Wer betreibt seinen +2A mit dem RGB-Ausgang an einem FS-Gerät oder Monitor und kann Aussagen machen, ob da gegenüber dem direkten COVO-FBAS-Signal noch wesentliche Verbesserungen zu erzielen sind? Wer kennt einen konkreten Fernsehgerätetyp, der mit dem +2A-RGB-Ausgang funktioniert?

Heinz Schober, Taubenheimer Str. 18
01324 Dresden

Vergoldeter Busanschluß (nicht nur für +D oder Disciple Besitzer)

Seid ihr +D oder Disciple Besitzer und plagt euch übermäßig oft mit Dingen wie 'Sektor Errors', Diskettendefekten oder unerklärlichen Abstürzen, vornehmlich in den ersten Minuten nach dem Anschalten? Nach meinen Erfahrungen ist der Grund sehr einfach in der verschmutzten Busleiste zu suchen. Das Reinigen der Busleiste hilft für eine Weile, aber das Problem kommt eventuell immer wieder vor.
Keine Sorge, es gibt Abhilfe. Was ihr braucht ist ein vergoldeter Busstecker, und den gibt es zu einem recht annehmbaren Preis (etwa 9 Pfund oder ca. 23 DM). Das Resultat ist wirklich excellent, und in meinem Fall das Geld wert. Der Busstecker wird einfach auf den existierenden gesteckt, doch zuerst muß man die Leiste mit einer Metallsäge auf die erforderliche Länge bringen, weil sie mehr Anschlüsse hat als der Spectrum Bus.
Vertrieben wird dies durch:

R.S. Components, P.O. Box 99
Corby, Northants NN17 9RS
England

Der Versandbetrieb von R.S.Components arbeitet übrigens mit einer 'Schwestergesellschaft' zusammen:

Electromail, P.O. Box 33
Corby, Northants NN17 9EL
England, Tel. GB-0536/201234.

Der Preis des aktuellen Kataloges beträgt 2,95 Pfund (ca. 7,50 DM).

Miles Kinloch, Flat 16, 6 Drummond Street
Edinburgh, EH8 9TU, Scotland, U.K.

FRAGEN ANZEIGEN

DTP und LPRINT III?
Ich habe folgendes Problem: Wie muß ich das Druckerinterface LPRINT III initialisieren, damit es einwandfrei mit den DTP-Programmen (Wordmaster / Headliner / Typeliner) zusammenarbeitet?
Ich habe wie im Handbuch des DTP vorgesehen den Befehl LPRINT' und zusätzlich CHR$ 5 (= Tokens off) eingegeben.
Leider hängt sich das ganze System beim Ausdruck auf oder druckt nur wirres Zeug.
Vielleicht kann hier jemand aus dem Club weiterhelfen. Mit besten Usergrüßen

Harald R. Lack, Heidenauer Straße 5
83064 Raubling

Sinclair Basic auf PC und Atari?
Mit einem einfachen Trick kann ich Sinclair-Basic in den Atari laden. Als ASCII-File.
Beispiel: Ramdisk einrichten. 1200 Baud wählen. ASCII wählen. Datei-Name wählen und nun Spectrum einschalten. Basic Programm (vom Band) laden. Eingeben: FORMAT 't',1200: CLEAR #: OPEN #2,'t': LIST, oder wenn's nicht soviel sein soll: PRINT PI.
Dies funktioniert mit dem Atari Programm "HBJ 86" seit 1986! Sinn der Sache? Mir fehlt es an Freizeit, immer zwischen Spectrum und Atari zu wählen. Einige wenige selber programmierte Programme gibt es immer noch nicht. Darum habe ich mich entschlossen, die Spectrum Programme auf dem Atari anzupassen. Die ersten Versuche mit GFA-Basic zeigten, daß dieses sehr umständlich war, da der Syntax zum Spectrum Basic total anders ist.
Nun habe ich mir das Atari Omikron Basic genauer angesehen. Und siehe da, es ist fast identisch mit dem Spectrum Basic. Nach der Übertragung vom Spectrum zum Atari empfehle ich, die ASCII Datei mit Tempus zu bearbeiten. Geht einfach schneller! Da kann man geschwind GO TO --> GOTO umändern und sonstige Anpassungen vornehmen. Danach kann der Rest direkt im Omikron Basic Interpreter verändert werden. Und siehe da... die Spectrum angepaßten Programme auf dem Atari laufen rasend schnell!
Nun suche ich ein PD Programm als Konverter, das die GFA und Omikron Basic Files konvertieren kann. Soll es laut Auskunft eines Freundes geben. Wenn ich nun die Omikron Programme in das GFA-Basic umwandeln kann sind wiederum die GFA-Basic Files mit dem MS-DOS GW-Basic oder GFA-Basic austauschbar. Nur die Window's Befehle fehlen, bzw. müssen auf dem Atari gelöscht werden.

Richard Raddatz, Pfarrgasse 5
71332 Waiblingen, Tel. 07151/563377

9 noch original verpackte 3-Zoll-Disketten für circa 50,- DM.
Spectrum +3 (eingebautes Laufwerk, externes Zweitlaufwerk, Druckerkabel, Handbuch und ca. 35 Arbeitsdisketten) ca. 290,- DM.
PC-Grafikkarte GMI0-470, VLB, 1MB, integrierter AT-Bus Controller und Schnittstellen ca. 90,- DM. **Dieter Schulze-Kahleuß, Tel. 07195/64404**

Wer kann Eurico helfen? Er sucht nach einem defekten SAM Coupe, einer CP/n Boot Diskette für PC, CP/M für PC, Spectrum Emulator für PC und MAC, sowie alle Arten von Konverterprogrammen, ebenfalls für PC und MAC. Ihr könnt ihm direkt in englisch oder portugiesisch schreiben oder eure Tips an uns geben, wir leiten sie dann weiter. Seine Adresse ist:

Eurico Oscar Covas, Rua das Milheiras, Lote 1
R/c Dto., 2080 Almeirim, Portugal

Biete ZX81 komplett, TV-Bild sehr verzerrt; ZX-Drucker mit Papierrollen, große Tastatur (Eigenbau) und Aufsatztastatur; QSave, einige Spiele. **Stefan Dressler, Jägerstraße 23a**
13595 Berlin, Tel. 030/3618723

Spectrum 48K+, gereinigt und getestet, komplett mit Handbuch, Netzteil und Kabeln 120,- DM; Spectrum 48K in DK'tronics-Tastatur, sehr stabil, mit eingebautem Kempston- und Sinclair-Joystickinterface, mit Netzteil, Handbuch und Kabeln 120,- DM; Spectrum 128K, gereinigt und getestet, komplett mit Netzteil, Kabeln und 128er Anleitung (Kopie) gegen Gebot.
Interface 2 gegen Gebot, Kempston-Joystickinterface 10,- DM; Wafer für Wafadrive, gebraucht, formatiert 4,- DM; Cartridges, gebraucht, formatiert, jeweils 10+1, 3,50 DM

Jean Austermühle, Postfach 10-1432
41546 Kaarst, Tel. 02131/69733

Verkaufe für 100 DM verbesserte nackte 48K Spectrum Platine. Gebrauchte und geprüfte IF1 ROM oder 48K Erweiterung für 16K Spectrum je 15 DM. **Richard Raddatz, Pfarrgasse 5**
71332 Waiblingen, Tel. 07151/563377

Verkaufe: Spectrum +2 (grau), Joystickports umgebaut, Ton und Bild am Fernseher optimiert 170,00 DM; Spectrum +2A wie +2, Busport umgebaut, ROM ausgetauscht 160,00 DM; Spectrum 48k (Gummi) 90,00 DM.
Joystickinterface 1-Port 10,00 DM, 2-Port 20,00 DM; U-Slot, verdoppelt Busport zum Anschluß weiterer Pheripherie 15,00 DM.

Thomas Eberle, Gastäckerstraße 23
70794 Filderstadt, Tel. 0711/777142